Brownian Motion Input Analysis for the Article Entitled: *Optimal Resource Allocation with Delay Guarantees for Network Slicing in Disaggregated RAN*

May 2025

Network Calculus is a mathematical framework, sometimes considered a special case of the queueing theory, that deals with bounds for backlog and delay in networked systems whose arrival and service processes are constrained by the so-called arrival and service curves. The theory is usually divided into two frameworks: Deterministic Network Calculus (DNC) and Stochastic Network Calculus (SNC) [1]. While the DNC framework addresses the worst-case scenario, the SNC allows bounds violations with specific probabilities. The former can be seen as a tool for network designers and managers to offer Quality of Service (QoS) guarantees in a strict sense. In contrast, the latter may provide guarantees in a wide sense, as it is always associated with probabilities of bound violations, with such violations occurring unpredictably in time and position across the network under analysis. The choice between DNC and SNC is a design decision, always tied to the network scenario and the problem under analysis. DNC has been widely used to solve problems in scenarios that demand determinism, where strict reliability and safety requirements are present. The following references are examples of the use of DNC in diverse communication and computing scenarios [2–8], among which we highlight [4], i.e., a work that emphasizes the use of DNC for URLLC, where we read:

> Worst-case delay is of paramount concern for 5G services like ultra-reliable low latency communication (URLLC), used for applications like robotic surgery, intelligent driving systems.

We also reference works that apply DNC to IEEE 802.1 TSN industrial scenarios and IETF DetNet environments [9–13]. Among these, we particularly highlight the contribution of [14], as these authors, including Le Boudec, made significant contributions to the development of DNC theory and its application to networks with strict QoS requirements.

The SNC has been used in several works [15–18] as the theoretical framework to estimate latency in 5G/B5G scenarios. Although not oriented toward specific applications, these works consistently assume that violations are allowed with a certain probability and that the stochastic behavior of network traffic is present in WAN TCP/IP networks or, more broadly, in Internet traffic. In this regard, we agree with the cited authors. However, such behavior is not observed in the industrial case of robot motion control and management, which is the focus of our article. This assertion is corroborated by several works in the literature, such as [4, 5, 9, 11].

We emphasize this point clearly. We are fully aware of the stochastic nature of network traffic in typical WAN TCP/IP networks, but this is not within the scope of our work. Our study addresses disaggregated RANs tailored for industrial networks. This work represents a bold research question: how can 5G and beyond technologies, particularly when using open architectures, drive the revolution of Industry 4.0 and beyond? That is precisely why we chose the deterministic framework. Changing our approach to the stochastic version of network calculus would establish our article on an incorrect premise, failing to address the problem of industrial applications under analysis and leading to misleading conclusions.

Moreover, it is fundamental to differentiate between the use of deterministic frameworks and the nature of the input traffic. Although industrial traffic profiles do not exhibit the stochasticity typically present in Internet traffic, our proposal, based on DNC, can still be applied to input traffic with stochastic characteristics, as in many cases, flows are shaped upon their admission to the network. This case is evident in TSN, which leverages traffic shaping to provide determinism across the network, and in the fact that many switches implement shaping mechanisms at the egress ports, making our approach fully compatible with practical deployments. In contrast, the practical deployment of SNC-based approaches remains an unresolved challenge, with limited recent progress.

Although we considered it essential to clarify the suitability of DNC to our work, we also understand that one would expect to see how the considered network premises would behave in the presence of random or stochastic traffic. In complete alignment with this suggestion, we considered the stochastic counterpart of the deterministic leaky-bucket shaper, the Brownian motion process, and its associated arrival curve with violation probability, as detailed in the following analysis.

In network traffic modeling, the arrival process $A(t)$, representing the cumulative traffic (in bits) that arrives up to time t, can be modeled as a linear term ρt plus a Brownian noise component $\delta W(t)$, as follows:

$$A(t) = \rho t + \delta W(t), \tag{1}$$

where ρ is the average arrival rate (in bits per second), δ is the standard deviation of the process (in bits per second), and $W(t)$ is a standard Wiener process satisfying:

$$W(t) \sim \mathcal{N}(0, t). \tag{2}$$

This model is appropriate when the arrivals exhibit variability or burstiness over time. As a result, the arrival process $A(t)$ is normally distributed with mean ρt and variance $\delta^2 t$:

$$A(t) \sim \mathcal{N}(\rho t, \delta^2 t). \tag{3}$$

In contrast to the leaky-bucket curve, which constrains the input network traffic using a linear expression $(\rho t + \sigma)$ based on the average arrival rate (ρt) plus a constant maximum burst (σ), we are interested in considering an arrival curve (envelope process) $\alpha(t)$ where the variability level changes over time. This leads to a stochastic approach to the envelope process, where the probability that the arrival process exceeds this envelope is bounded by a small parameter ε:

$$\mathbb{P}(A(t) > \alpha(t)) \leq \varepsilon. \tag{4}$$

To achieve this, we apply a probabilistic inequality, known as Chernoff's bound, which is particularly effective for bounding the tails of statistical distributions, and especially tight for Gaussian random variables.

Let us define the deviation $\Delta\alpha$ as the amount by which $\alpha(t)$ exceeds its mean:

$$\alpha(t) = \rho t + \Delta\alpha \quad \Rightarrow \quad \Delta\alpha = \alpha(t) - \rho t. \tag{5}$$

Then, the probability of envelope violation can be expressed as:

$$\mathbb{P}(A(t) > \rho t + \Delta\alpha) = \mathbb{P}(A(t) - \rho t > \Delta\alpha). \tag{6}$$

Since $A(t) - \rho t \sim \mathcal{N}(0, \delta^2 t)$, we have a zero-mean Gaussian variable, for which Chernoff's bound provides the following inequality:

$$\mathbb{P}(A(t) > \rho t + \Delta\alpha) \leq \exp\left(-\frac{(\Delta\alpha)^2}{2\delta^2 t}\right), \tag{7}$$

and

$$\exp\left(-\frac{(\Delta\alpha)^2}{2\delta^2 t}\right) = \varepsilon \quad \Rightarrow \quad -\frac{(\Delta\alpha)^2}{2\delta^2 t} = \log(\varepsilon). \tag{8}$$

Isolating $\Delta\alpha$ in (8), we find:

$$\Delta\alpha = \delta\sqrt{2t\log\left(\frac{1}{\varepsilon}\right)}. \tag{9}$$

Therefore, the arrival curve $\alpha(t)$ that bounds the arrival process with probability $1 - \varepsilon$ is given by:

$$\alpha(t) = \rho t + \delta\sqrt{2t\log\left(\frac{1}{\varepsilon}\right)}, \tag{10}$$

which can be rewritten as:

$$\alpha(t) = \rho t + \kappa\delta\sqrt{t}, \tag{11}$$

where $\kappa = \sqrt{-2\log(\varepsilon)}$. Regarding the comparison between the leaky-bucket curve and the Brownian motion process, assuming that they share the same average arrival rate ρ, we can write:

$$\rho t + \sigma = \rho t + \kappa\delta\sqrt{t}. \tag{12}$$

Subtracting ρt from both sides yields:

$$\sigma = \kappa\delta\sqrt{t}, \tag{13}$$

and solving for t gives the intersection point in time for both arrival curves, which we refer to as the **time of interest** ($t.o.i.$), denoted by t^*:

$$t^* = \left(\frac{\sigma}{\kappa\delta}\right)^2. \tag{14}$$

This result expresses the critical point in time at which the stochastic envelope intersects the deterministic one. For all $t < t^*$, the leaky-bucket envelope dominates, being more conservative, whereas for $t > t^*$, the Brownian envelope becomes the tighter upper bound, although with a probability ε of violation.

It leads to the conclusion that for all $t < t^*$, there will be a curve with mean ρt and burst σ that upper bounds the arrival curve with mean ρt, standard deviation δ, and violation probability ε.

In the following, we present two scenarios to exemplify the obtained results using our approach. Table 1 summarizes the parameters used for both scenarios.

Table 1: Simulation parameters used for traffic generation

Parameter	**Scenario 1**	**Scenario 2**	**Description**
ρ	1×10^6 bps	1×10^6 bps	Average traffic rate
σ	1.024×10^3 bits	2×10^5 bits	Leaky-bucket burst
δ	4.45×10^2 bps	5.12×10^4 bps	Standard deviation of traffic
ε	0.005	0.005	Envelope violation probability
Δt	0.001 s	0.001 s	Time step resolution
`time`	2.5 s	2.5 s	Total simulation duration
`runs`	80	80	Number of traffic flows

In Scenario 1, we obtained δ using (26) and the parameters in Table 1. In other words, we used the industrial use case explored in our article (i.e., robot motion control with strict latency requirement) to derive the network traffic characteristics and generate stochastic traffic. As shown in Figure 1, the Brownian envelope closely matches the leaky-bucket envelope. The zoomed-in portion of the figure helps illustrate that the 80 runs (referring to 80 robots) produced

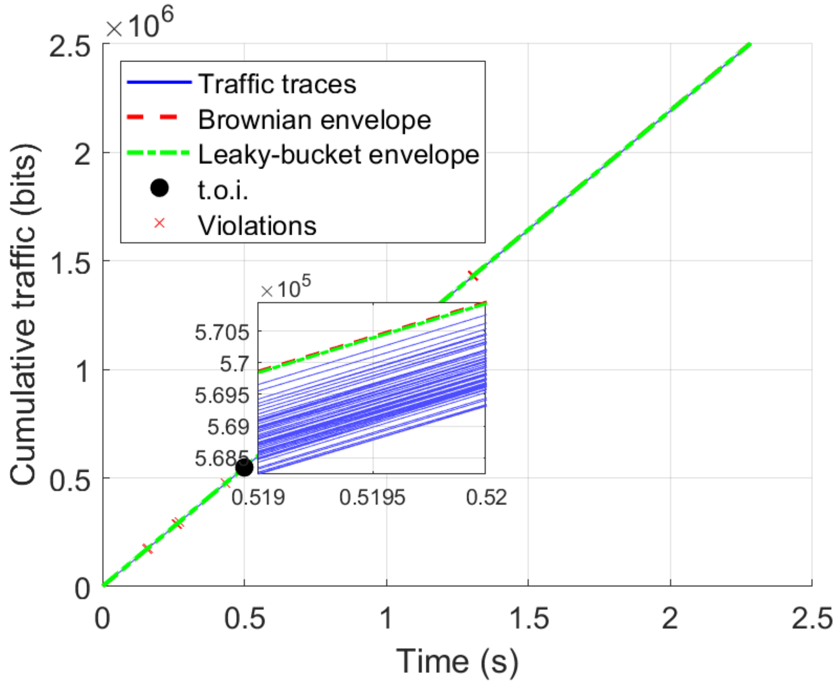

Figure 1: Scenario 1: Stochastic Traffic Profile expected for Motion Control in Industry 4.0

80 distinct traffic traces, but with curves that almost completely overlap. The figure also highlights, using a black dot, the time of interest (t.o.i.) and marks in red the violations of the Brownian envelope (with a probability of 0.5%).

To stress the network simulation further, we emulated a more variable network traffic profile in Scenario 2, depicted in Figure 2, by increasing the network traffic standard deviation, as seen in Table 1. As a consequence, the t.o.i. point changes, as well as the behavior of the blue curves. Although this behavior is not expected for network traffic in a factory robot motion control system, we varied the standard deviation deliberately to observe its impact on the network simulation using OMNeT++.

In Figure 3, we see that in Scenario 1 (representative of the industrial case), the latency simulated using the stochastic traffic closely matches the deterministic one. In Scenario 2, we observe that the stochastic traffic with artificially increased variability presents greater latency than the deterministic traffic, which is expected, as the stochastic envelope allows violations to the bound with a certain probability (0.5% in our case).

In summary, with a network traffic profile exhibiting increased variability, the OMNeT++ simulations showed that latency tends to be greater when fed with stochastic traffic than its deterministic counterpart. However, when we con-

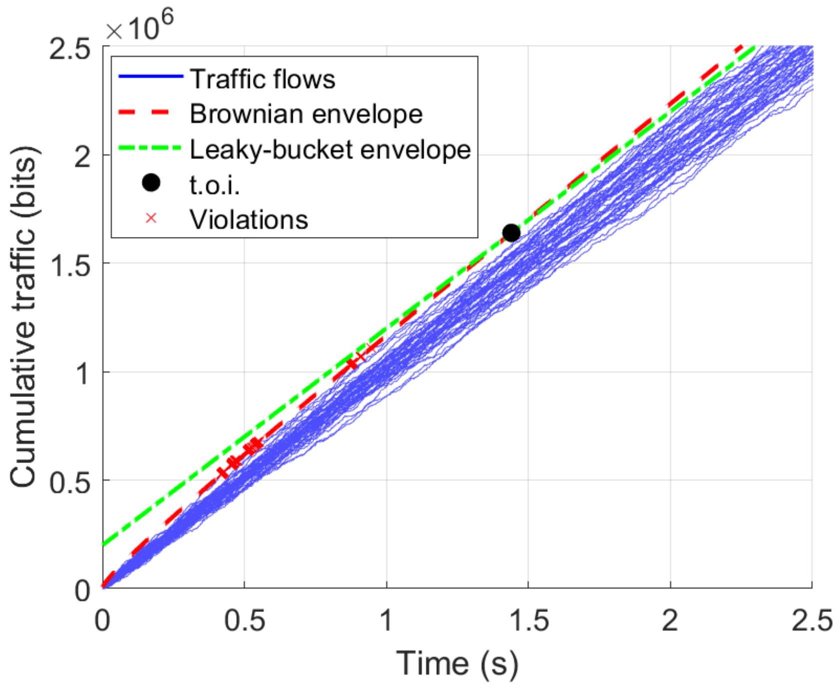

Figure 2: Scenario 2: Stochastic Traffic Profile with increased variability

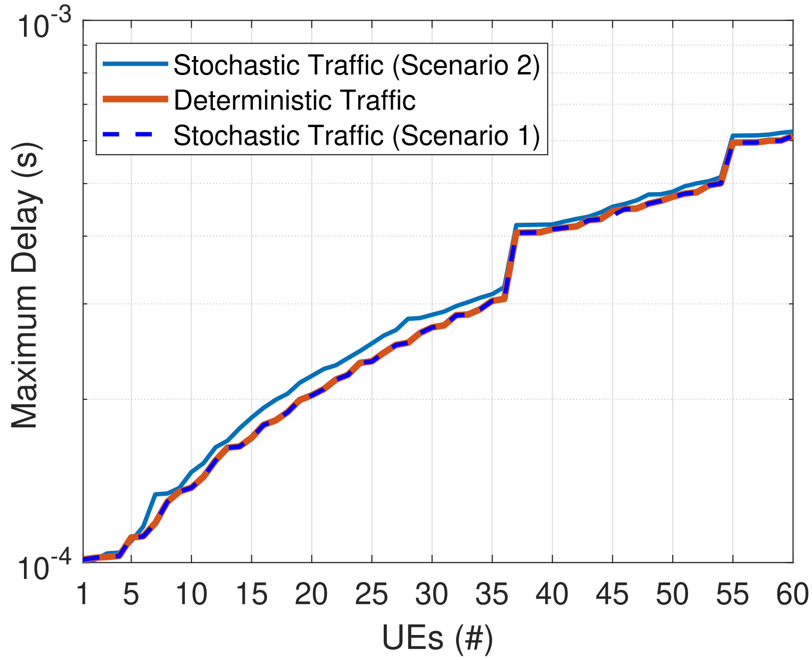

Figure 3: Maximum Delay per Number of UEs in a Simulation of the Network Scenario in OMNeT++

sider the network traffic characteristics of the industrial cases, as demonstrated in [4, 19], the δ parameter (the standard deviation of the Brownian motion process) is significantly reduced, and no meaningful difference can be observed between using the stochastic envelope and the deterministic envelope. Therefore, for Industry 4.0, both for analytical purposes and practical deployments, the leaky-bucket shaper remains a better recommendation than the stochastic counterpart.

Limitation: We clarify that deriving an upper bound for the stochastic envelope does not yield the delay violation probability directly. Obtaining such a probability would require using tools such as Large Deviation Theory (LDT) and Moment Generating Function (MGF) techniques. Nevertheless, this is beyond the scope of the present article, which focuses on the industrial case. The proposed envelope-based approach can be applied to realistic scenarios with deterministic traffic, as typically observed in robotic arms, Automated Guided Vehicles (AGVs), and other IoT devices. We also highlight that our proposal adheres to practical deployments, leveraging traffic shapers to handle eventual traffic variability, as commonly implemented in TSN and DetNet networks. As such, although theoretically sound, the use of SNC falls outside the scope of this work.

References

[1] M. Fidler, "Survey of deterministic and stochastic service curve models in the network calculus," *IEEE Communications surveys & tutorials*, vol. 12, no. 1, pp. 59–86, 2010.

[2] H. Wang and Y. Zhang, "Deterministic data center network calculus," in *2024 IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA)*. IEEE, 2024, pp. 1177–1182.

[3] Y. Zhang, C. Peng, S. Xie, and X. Du, "Deterministic Network Calculus-based H∞ Load Frequency Control of Multiarea Power Systems under Malicious DoS Attacks," *IEEE Transactions on Smart Grid*, vol. 13, no. 2, pp. 1542–1554, 2021.

[4] A. Mathew, M. Srinivasan, and C. S. R. Murthy, "Network calculus based delay analysis for mixed fronthaul and backhaul 5G networks," in *IEEE 21st international symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*. IEEE, 2020, pp. 205–214.

[5] Q. Wang, Z. Mo, B. Yin, L. Zhang, and P. Dong, "Bounding the upper delays of the tactile internet using deterministic network calculus," *Electronics*, vol. 12, no. 1, p. 21, 2022.

[6] S. Ma, S. Wang, and W.-T. Tsai, "Delay analysis of consensus communication for blockchain-based applications using network calculus," *IEEE Wireless Communications Letters*, vol. 11, no. 9, pp. 1825–1829, 2022.

[7] K. Feng and F. Baccelli, "Spatial network calculus and performance guarantees in wireless networks," *IEEE Transactions on Wireless Communications*, vol. 23, no. 5, pp. 5033–5047, 2023.

[8] R. Zippo and G. Stea, "Computationally efficient worst-case analysis of flow-controlled networks with network calculus," *IEEE Transactions on Information Theory*, vol. 69, no. 4, pp. 2664–2690, 2023.

[9] F. Geyer and S. Bondorf, "Network synthesis under delay constraints: The power of network calculus differentiability," in *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*. IEEE, 2022, pp. 1539–1548.

[10] J. Guo, H. Wang, Z. Wang, and Z. Chang, "Upper bound analysis of tsn end-to-end delay based on network calculus," in *2023 5th International Conference on Natural Language Processing (ICNLP)*. IEEE, 2023, pp. 394–399.

[11] M. Seliem, A. Zahran, and D. Pesch, "Delay analysis of tsn based industrial networks with preemptive traffic using network calculus," in *2023 IFIP Networking Conference (IFIP Networking)*. IEEE, 2023, pp. 1–9.

[12] J. Miserez, G. P. Sharma, and W. Tavernier, "Routing protocols exploiting queue information for deterministic networks," in *2023 19th International Conference on the Design of Reliable Communication Networks (DRCN).* IEEE, 2023, pp. 1–8.

[13] J. Miserez, D. Colle, M. Pickavet, and W. Tavernier, "Exploiting queue information for scalable delay-constrained routing in deterministic networks," *IEEE Transactions on Network and Service Management*, 2024.

[14] E. Mohammadpour, E. Stai, and J.-Y. Le Boudec, "Improved network-calculus nodal delay-bounds in time-sensitive networks," *IEEE/ACM Transactions on Networking*, vol. 31, no. 6, pp. 2902–2917, 2023.

[15] O. Adamuz-Hinojosa *et al.*, "A Stochastic Network Calculus (SNC)-based model for planning B5G uRLLC RAN slices," *IEEE Transactions on Wireless Communications*, 2022.

[16] Y. Han, M. Yao, J. Yang, Y. Luo, and X. Liu, "On scalable delay bound evaluation with stochastic network calculus," in *2020 IEEE Symposium on Computers and Communications (ISCC).* IEEE, 2020, pp. 1–6.

[17] S. Ma, X. Chen, Z. Li, and Y. Chen, "Performance evaluation of urllc in 5g based on stochastic network calculus," *Mobile Networks and Applications*, vol. 26, no. 3, pp. 1182–1194, 2021.

[18] O. Adamuz-Hinojosa *et al.*, "ORANUS: Latency-tailored Orchestration via Stochastic Network Calculus in 6G O-RAN," *arXiv preprint arXiv:2401.03812*, 2024.

[19] A. E. Kalør *et al.*, "Network slicing in industry 4.0 applications: Abstraction methods and end-to-end analysis," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 12, pp. 5419–5427, 2018.